

Nic Melder PhD, BSc (Hons), MIET

1113-191 Sherbourne Street, Toronto, Ontario, M5A 3X1, Canada

Telephone (mobile): +1 647 918 4737

Email: linkedin@riviera76.fastmail.fm, Web page: <http://cv.nicmelder.fastmail.fm>

Key Skills/Experience

- 14+ years experience developing vehicle AI systems for open world and racing games.
- 2 years experience developing physics systems for racing games (including vehicle dynamics and collision detection and response).
- 9+ years experience leading a multi-disciplinary team that was comprised of AI, physics, animation and systems programmers numbering 7 people at its peak.
- C++ and object oriented design and programming for PC, Xbox and Playstation.
- Experience in profiling and optimising for memory and CPU including SIMD vectorisation.
- Tools development using C#, WPF, MFC and MAXScript for multi-threaded, distributed automation applications, general tools and 3D Studio Max tools.
- Experience working on large commercial projects with 100+ programmers and budgets in excess of £50 million (10 million+ lines of code).
- Full project life cycle experience from design to release including post release support.
- Strong analytical and design skills.
- Strong technical writing skills including academic writing.
- 14 published games (2 at Ubisoft, 10 racing games at Codemasters, 2 at Asylum Entertainment).
- Five published articles related to artificial intelligence in racing games ('Game AI Pro: Collected Wisdom of Game AI Professionals', September 2013).
- Invited speaker at the Game/AI Conference, Vienna, 2012, talking about racing AI.
- Six academic papers and thesis published in the field of VR and Haptics.
- Presented at the three leading Haptics conferences and demonstrated my Multi-finger Haptic system at various public events / conferences (including at the Science Museum in London).

Employment History

Vehicle AI Programmer (Lead) (Level 4 / Principal)

Ubisoft Toronto, 224 Wallace Ave, Toronto, ON M6H 1V7, Canada

June 2016 - Present

As the technical and team lead for the vehicle AI team on Watchdogs: Legion, my primary responsibilities were to design, develop and direct the development of the new vehicle AI (traffic and felony/chase) systems for Watchdogs: Legion. I was also responsible for updating and enhancing the road network data systems and exporters and converting it from being a US based system (mainly grid based, driving on the right) to a UK based system (driving on the left, roundabouts, much more complex intersections).

I also managed a team of two other programmers and was responsible for planning and documentation on top of actually implementing and delivering the new systems. As the lead, I was the primary point of contact for all things vehicle AI and regularly updated the senior project leadership (lead programmers, creative and game directors, and senior production) on the current state and progress of the vehicle AI systems.

I have also spent time assisting Watchdogs 2, and Far Cry 6 with vehicle AI issues whilst their projects were being finalised for release.

Some of the notable features of the new vehicle AI system that I designed and developed follows:

Single, unified vehicle AI system - It was decided to replace the multiple vehicle AI systems from Watchdogs 2 with a single unified architecture. Watchdogs 2 had two completely separate systems for normal traffic and police driving / chases, a different system for boat traffic, and a fourth system for the helicopters. The new system was used for traffic (vehicle and boats), the high speed chases and also for the new drone system. This simplified the code by removing around 300 files and also meant that if a vehicle's handling or characteristics changed then it would only need to be changed in a single place in the code to make it drive correctly. The high speed chases, would use the exact same code as the normal traffic, except that it would employ additional driving behaviours (overtake) and driving characteristics (greater desired acceleration / braking, no road speed limits and could ignore traffic lights and pedestrians). A high speed felony vehicle was also much more processor performant i.e. a single high speed vehicle in the new system took under 100us whereas in the old system, it took more than 800us.

Thread friendly and Engine agnostic - The new Vehicle AI system was designed to be as engine agnostic as possible. Access to the systems were through well defined interfaces so that it should be relatively straightforward to transplant this system to a new engine as only the external interfaces would need to be re-implemented. Each vehicle would also internally cache all the data it needed in order to determine it's behaviour using the external interfaces. This had the added advantage that they could then be processed independently of the other systems and on a different thread if required.

Upgraded road network system - The road network represents the roads / sidewalks and how they connect together. Watchdogs 1 and 2 were both set in the US and were designed for a grid based road layout and driving on the right. Watchdogs: Legion required that the vehicles drove on the left and included different junction types (4 lane roundabouts, multiple types of pedestrian crossings) and much more complicated intersections (6 way traffic light intersections, 5 way stop intersections, complex forks and merges (e.g. a 6 lane road forking/merging into 3 separate lanes)). The final upgraded system now supports these exotic junction types and now supports both left and right hand driving systems.

Principal Programmer: Artificial Intelligence and Physics, Group Lead

Codemasters Software Company Ltd, Southam, Warwickshire, CV47 2DL, England

October 2006 – February 2015

My primary responsibilities were to design, develop and enhance the AI and Physics systems for use in the Racing Studio's critically acclaimed AAA racing titles for PC and consoles. Across the projects that I have worked on (DiRT, GRID, DiRT2, F1 2010, DiRT 3, DiRT Showdown, GRID 2, GRID:Autosport, DiRT:Rally and F1 2015), I have been responsible for planning, documenting and implementing any new AI features required. From May 2012, I was also directly responsible for implementing any new Physics systems. This also included the creation, development and maintenance of the AI and Physics specific tools and build chains using C#, WPF, MFC and MAXScript. I was also the primary point of contact between the level and car handling design teams (17 people) and the programming team as well as answering to senior management (lead programmers, chief designers and executive producers) with regard to anything related to the AI or Physics in the games.

Since 2013, a large part of my responsibilities included working with the F1 team to design and implement the next gen (EGO3) AI and Physics systems such that a common code base could be shared between the F1 studio and the Racing studio.

Some recent notable features that I have been responsible for follow. In all cases I was responsible for both developing and managing any other programmers that were involved, as well as presenting the results to senior management for sign off.

Breakable Objects (F1 2015/EGO3 Engine) – As the sole programmer, I worked closely with the artists to design and implement a completely new system for creating objects that would deform and / or break apart when collided with for Codemasters' next gen engine (EGO3). This required a complete redesign of how the objects were to be rigged in 3D Studio Max, a set of MAXScript tools for rigging and converting the legacy assets to the new format, writing a new build pipeline for all the physics objects (in C#) as well as the game code for activating, hiding and resetting the collideable objects (in C++). The new system had approximately 90% of the functionality that the previous system had (the 10% missing wasn't required for F1) yet was more flexible and much simpler to understand by the artists and programmers (the artist setup guide for the new system was 18 pages (including examples) vs 80 pages). Due to the simplicity of the new breakables system, the Racing Studio is planning on adopting it for use in its next title.

Arena Based AI (DiRT Showdown) – With only 5 months for development, working with the level and game designers and another AI programmer, this involved developing an entirely new AI system in order for the AI to actively target / avoid vehicles and fixed objects in open world arenas for multiple destruction derby type game modes. This required automatic spline generation, fixed and dynamic obstacle avoidance, a new behaviour system, a high level AI director for global decision making (who should target who, when to run away, how many to attack the player etc.) and a realistic difficulty system where cars would purposely miss the player / each other instead of just applying score / damage multipliers.

Research Fellow in Multi-Point Haptics

*Department of Cybernetics, University of Reading, Reading, RG6 6AY, England
October 2001 - October 2006 (Part-time until April 2002)*

AI Programmer - Asylum Entertainment Ltd, London, England
September 2000 - April 2002 (Part-time from Oct 2001)

Computer Analyst/Programmer - Sunray Communications, Southend, Essex, England
September 1996 - September 2000 (Part-time)

Computer Analyst/Programmer - HSBC Bank PLC, London, EC3R 6HA, England
October 1995 - October 1997

Education

PhD in Multi-Point Haptics - University of Reading, England
January 2002 - June 2011 (Part-time)

BSc (Hons) Cybernetics and Control Engineering (2.i) - University of Reading, England
September 1997 - June 2000